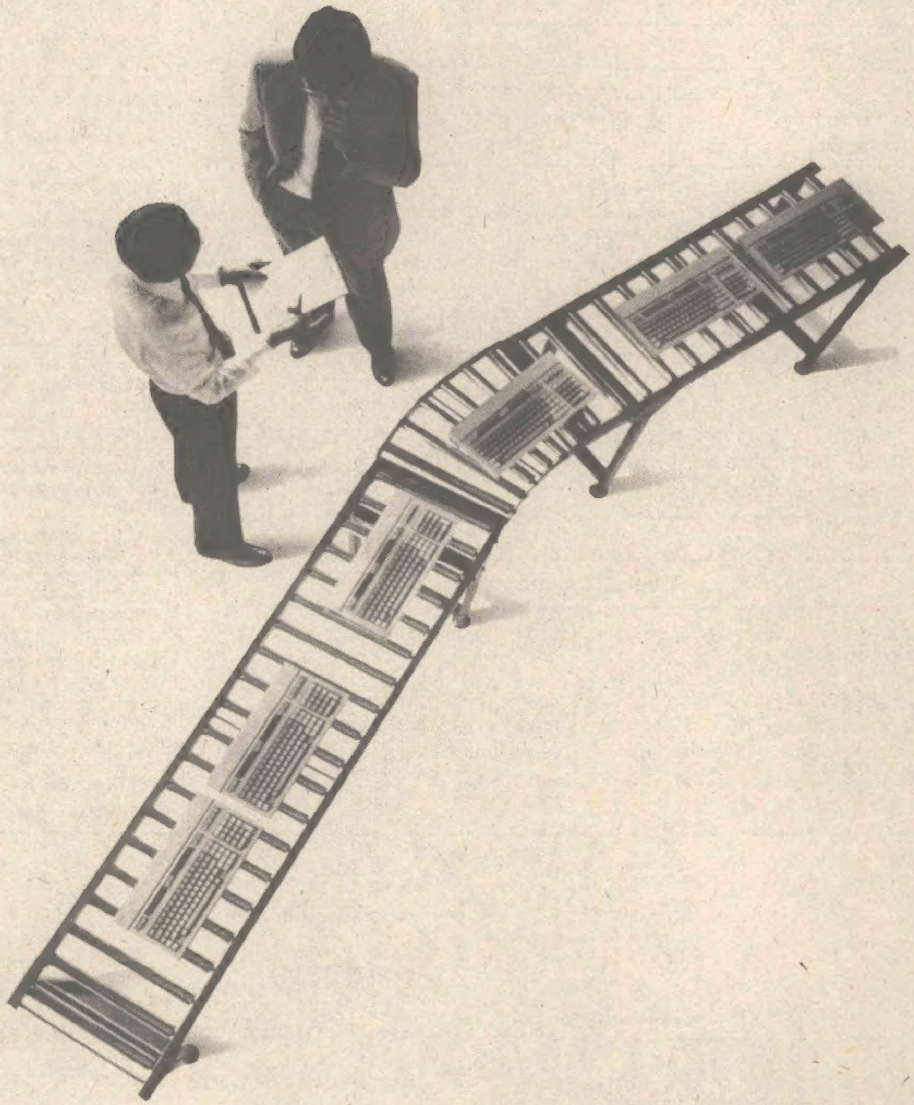
สรรค์ เสกขุนทด

วินโดวส์โปรแกรมมิง : Controls

แนะนำแบบของคอนโทรล และการใช้งาน ในยามปกติวิสัย



วินโดวส์โปรแกรมมิงสำหรับฉบับนี้จะเป็นเรื่องคอนโทรลส์ (controls) ซึ่งเป็นส่วนประกอบสำคัญของไดอะล็อกบ็อกซ์ที่ผู้อ่านคงจะคุ้นเคยกันดีอยู่แล้ว เพราะโปรแกรมบนวินโดวส์ทั้งหลายจะใช้ไดอะล็อกบ็อกซ์ในการติดต่อกับผู้ใช้ เช่น เปิดไฟล์ พรินต์ไฟล์ หรือถ้าเป็นโปรแกรมเกมก็จะมีไดอะล็อกบ็อกซ์สำหรับออปชันหรือแสดงรายชื่อผู้เล่นได้คะแนนสูงสุดเป็นต้น

เข้าเรื่อง

ก่อนที่จะเข้าสู่เรื่องคอนโทรลส์นั้นก็ขออธิบายเกี่ยวกับ วินโดวส์สไตล์ (window style) เสียก่อนเพื่อจะได้เข้าใจได้ลึกซึ้งยิ่งขึ้น

ตามปกตินั้นแต่ละแอพพลิเคชันจะมีวินโดวส์หลักอยู่หนึ่งวินโดวส์อยู่แล้ว เพื่อใช้เป็นที่แสดงผลคืออะไรก็แล้วแต่ วินโดวส์หลักนี้มักจะประกอบด้วยไตเติ้ลบาร์, ซิสเต็มเมนู, มินิไมซ์บ็อก, แมกซิไมซ์บ็อก และเมนู เป็นต้น ถ้าท่านผู้อ่านยังจำได้ จะเห็นว่าเราจะสร้างวินโดวส์หลักนี้โดยเรียกฟังก์ชัน CreateWindow ซึ่งจะมีพารามิเตอร์ที่บอกเป็นวินโดวส์สไตล์ สำหรับวินโดวส์หลักเรามักจะใช้ WS_OVERLAPPED-WINDOW เป็นวินโดวส์สไตล์

ชนิดของวินโดวส์

วินโดวส์มีทั้งหมด 4 ชนิดคือ

1. โอเวอร์แล็ปวินโดวส์ (Overlapped Windows)

มักใช้เป็นวินโดวส์หลักของแอพพลิเคชัน วินโดวส์ชนิดนี้ประกอบด้วยไคลเอ็นต์แอเรีย, บอร์เดอร์ และไตเติ้ลบาร์ นอกจากนี้ก็อาจจะมีส่วนประกอบอื่น ๆ อีกเช่น ซิสเต็มเมนู มินิไมซ์บีอก เป็นต้น

2. โอนว์วินโดวส์ (Owned Windows)

เป็นชนิดพิเศษของโอเวอร์แล็ปวินโดวส์ วินโดวส์ชนิดนี้จะมี "เจ้าของ" (Owner) เป็นโอเวอร์แล็ปวินโดวส์เสมอ และมีคุณสมบัติดังนี้

- อยู่เหนือเจ้าของเสมอ
- ถูกทำลายโดยอัตโนมัติถ้าเจ้าของถูกทำลาย
- เมื่อเจ้าของถูกมินิไมซ์จะถูกซ่อนเอาไว้

ไดอะล็อกบ็อกซ์ทั้งหลายเป็นวินโดวส์ชนิดนี้

3. ป๊อปอัพวินโดวส์ (Pop-up Windows)

เป็นชนิดพิเศษของโอเวอร์แล็ปวินโดวส์เช่นกัน มีข้อแตกต่างกันเล็กน้อยคือวินโดวส์ชนิดนี้อาจจะมีหรือไม่มีไตเติ้ลบาร์ก็ได้ ส่วนโอเวอร์แล็ปวินโดวส์จะมีเสมอ

4. ไชด์วินโดวส์ (Child Window)

ไชด์วินโดวส์เป็นวินโดวส์ที่จะต้องมีวินโดวส์พ่อ (parent) เสมอ และจะอยู่ภายในไคลเอ็นต์แอเรียของวินโดวส์พ่อเท่านั้น ดังนั้นจึงถูกใช้ทำหน้าที่แบ่งพื้นที่ของไคลเอ็นต์แอนเรียออกเป็นส่วน ๆ เพื่อให้ทำหน้าที่แตกต่างกันไป

วินโดวส์พ่อสามารถเป็นวินโดวส์ชนิดใดก็ได้ รวมทั้งไชด์วินโดวส์ นอกจากที่กล่าวมาแล้ว ไชด์วินโดวส์มีคุณสมบัติเหมือนกับวินโดวส์ชนิดอื่น ๆ

สำหรับรายละเอียดของวินโดวส์สไตล์นั้น หาอ่านได้จาก Reference ของ SDK

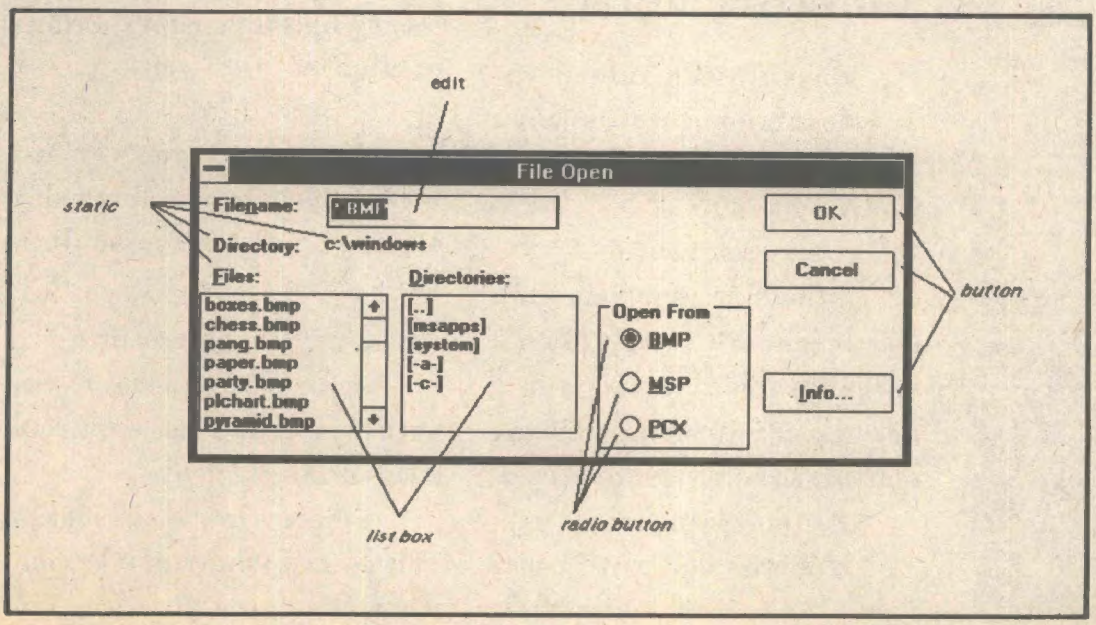
ค้นหาความลับด้วยสปาย (spy)

ใน SDK 3.0 และ 3.1 มีโปรแกรมหนึ่งชื่อว่า spy เป็นโปรแกรมใช้ในการดูเมสเสจที่เกิดขึ้นระบบ เราสามารถเลือกดูเมสเสจที่ส่งมาให้วินโดวส์ใด ๆ ได้โดยการเลือกเมนู Window จะปรากฏไดอะล็อกบ็อกซ์เพื่อให้เลือกวินโดวส์ที่ต้องการดูเมสเสจ จากนั้นให้เลื่อนเม้าส์ไปที่ส่วนของวินโดวส์ที่ต้องการแล้วคลิ๊ก OK ช่วงที่เลื่อนไปนั้นจะปรากฏข้อมูลต่าง ๆ ของวินโดวส์ ซึ่งจะมีวินโดวส์ไดล์ด้วย ดังนั้นเราจึงสามารถใช้โปรแกรมสปายในการดูว่าเขาออกแบบแอพพลิเคชันบนวินโดวส์กันอย่างไร

ว่าด้วยคอนโทรล

ดังที่ได้กล่าวไปแล้วว่า คอนโทรลเป็นส่วนประกอบสำคัญของไดอะล็อกบ็อกซ์ ขอให้ผู้อ่านดูรูปที่ 1 ซึ่งแสดงไดอะล็อกของ pbrush เพื่อใช้เปิดไฟล์ จากรูปจะเห็นคอนโทรลหลาย ๆ ชนิดเพื่อใช้ในการติดต่อกับผู้ใช้

แล้วคอนโทรลจริง ๆ แล้วคืออะไร ถ้าผู้อ่านลองใช้ spy จึงดูจะพบว่าคอนโทรลก็คือไชด์วินโดวส์นั่นเอง เพียงแต่เราไม่ต้องสร้าง วินโดวส์ฟังก์ชันเอง



รูปที่ 1
ไดอะล็อกบ็อกซ์ open file ของ paint brush

โดยระบบวินโดวส์ได้เตรียมไว้ให้ใช้แล้ว เราเพียงแต่นำมาประกอบกันเท่านั้นเอง ทำให้พัฒนาโปรแกรมได้รวดเร็วและโปรแกรมมีลักษณะใช้งานง่าย

ชนิดของคอนโทรล

วินโดวส์ได้เตรียมคอนโทรลไว้ให้ใช้ทั้งหมด 6 ชนิด แต่ละชนิดมีหลากหลายสไตล์ด้วยกันคือ

1. button

ก็คือปุ่มนั่นเอง ปุ่มก็มีหลายชนิด เช่น ปุ่มธรรมดา ปุ่มแบบกลม ปุ่มแบบสี่เหลี่ยม เป็นต้น

2. Edit

เป็นที่ใส่ข้อความต่าง ๆ เช่น ชื่อไฟล์ เป็นต้น

3. List-box

เป็นที่แสดงรายการต่าง ๆ เช่น ชื่อไฟล์ ชนิดของไฟล์ เป็นต้น

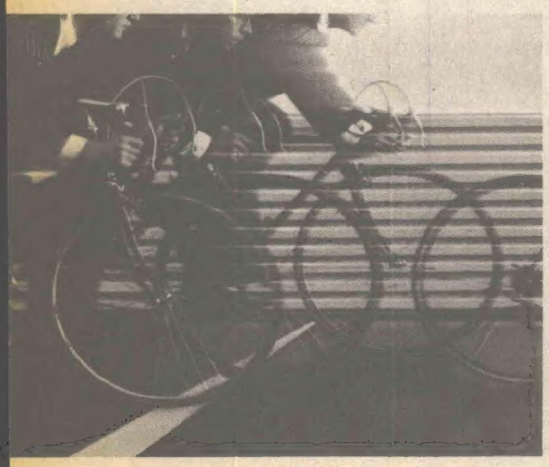
4. Scroll-bar

ใช้ในการเลื่อนดูข้อมูล ในกรณีที่มีข้อมูลมากจนไม่สามารถแสดงไว้ภายในหน้าเดียวได้

5. Static

ใช้แสดงข้อความที่ไม่มีการเปลี่ยนแปลง เช่น บอกว่าช่อง edit นี้ใช้ทำอะไร

6. Combo-box



คอนโทรลส์	ชื่อคลาส
button	BUTTON
edit	EDIT
list box	LISTBOX
combo box	COMBOBOX
scroll bar	SCROLLBAR
static	STATIC

รูปที่ 2 แสดงความสัมพันธ์ระหว่างคอนโทรลกับชื่อคลาส

```
HWND hButton;

hButton = CreateWindow (
            "Button",                                  /* ชื่อคลาส     */
            "OK",                                      /* ชื่อ button */
            BS_PUSHBUTTON | WS_CHILD | WS_VISIBLE,     /* สไตล์        */
            20,                                        /* ตำแหน่ง x */
            40,                                        /* ตำแหน่ง y */
            30,                                        /* ความกว้าง    */
            12,                                        /* ความยาว      */
            hWnd,                                      /* วินโดว์พ่อ */
            IDOK,                                      /* หมายเลข      */
            hInstance,                                 /* อินสแตนต์ */
            NULL);
```

รูปที่ 3 ตัวอย่างการสร้างคอนโทรล

เป็นคอนโทรลที่เป็นส่วนผสมของ edit และ listbox

วิธีการสร้างคอนโทรล

เมื่อคอนโทรลเป็นวินโดวส์ ดังนั้นจึงต้องสร้างคอนโทรลด้วยฟังก์ชัน CreateWindow นั่นเอง โดยมีพารามิเตอร์ที่สำคัญดังนี้

1. คลาสของวินโดวส์

วินโดวส์ได้กำหนดมาแล้ว ดังนั้นเราจึงใส่ชื่อคลาสให้สอดคล้องกับชนิดของคอนโทรลที่ต้องการสร้าง รูปที่ 2 แสดงความสัมพันธ์ระหว่างชนิดของคอนโทรลและชื่อคลาสของคอนโทรล

2. สไตล์ของคอนโทรล

เราต้องระบุลงไปว่าเราต้องการปุ่มแบบใด edit box มีลักษณะอย่างไร ซึ่งเรื่องนี้จะกล่าวอีกในรายละเอียดแต่ละคอนโทรล

เนื่องจากว่าคอนโทรลเป็นไชด์วินโดวส์ ดังนั้นจึงต้องระบุ WS_CHILD ลงไปในสไตล์ด้วย

3. วินโดวส์พ่อ

เนื่องจากคอนโทรลเป็นไชด์วินโดวส์ ดังนั้นต้องสร้างคอนโทรลซึ่งต้องระบุเสมอว่า วินโดวส์พ่อของคอนโทรลคือวินโดวส์ใด

4. หมายเลขของคอนโทรล

ระบุโดยใส่ในพารามิเตอร์ hMenu ของฟังก์ชัน CreateWindow หมายเลขนี้ไว้อ้างอิงถึงคอนโทรลได้

ตัวอย่างการสร้างคอนโทรลดูได้จากรูปที่ 3 ซึ่งเป็นการสร้างโดยเรียกผ่านฟังก์ชัน CreateWindows วิธีนี้

สามารถสร้างคอนโทรลบนวินโดวส์ได ๆ ก็ได้ สำหรับวิธีการสร้างคอนโทรลในไดอะล็อกบ็อกซ์นั้นผู้อ่านติดตามได้ในฉบับต่อไป

เจาะลึก

มาถึงบรรทัดนี้แล้ว ท่านผู้อ่านก็คงพอจะเข้าใจเจ้าคอนโทรลขึ้นมาบ้าง ต่อไปเราจะไปศึกษาคอนโทรลแต่ละชนิดให้ละเอียดยิ่งขึ้น

Button

button เป็นคอนโทรลที่ใช้รับอินพุตจากผู้ใช้ในลักษณะ yes/no, on/off และรับคำสั่งให้ทำตามที่ต้องการ สไตล์ของ button มีหลายสไตล์ให้เลือกใช้ตามเหมาะสมกับงาน

Push Button

เป็น button เพื่อให้ผู้ใช้เลือกทำงานสักอย่าง ตัว button จะมีข้อความบอกว่า button ใช้ทำอะไรเช่น SAVE, OK, CANCEL, EXIT เป็นต้น

สำหรับการสร้าง button ทำได้โดยเรียกฟังก์ชัน CreateWindows โดยมีเป็นสไตล์เป็น BS_PUSHBUTTON

Default Push Buttons

ทำงานคล้าย ๆ กับ Push Button แต่มีสไตล์เป็น BS_DEFPUSHBUTTON

Check Boxes

ใช้ในงานในลักษณะที่ให้ผู้ใช้เลือกกลุ่มของออปชันไว้มากกว่าหนึ่งทาง ยกตัวอย่างเช่น ต้องการแก้ไขลักษณะตัวอักษรให้เป็นตัวหนา, ตัวเอียง หรือตัวหนาเอียง ในกรณีนี้ใช้ check box สำหรับตัวหนาและตัวเอียง เป็น 2 check box แล้วให้ผู้ใช้เลือกอันใดอันหนึ่งหรือทั้งสองอันได้

สไตล์ของ check-box คือ BS_CHECKBOX สำหรับการสั่งให้ check_box แสดงตัวเช็ค (X) ทำได้โดยส่งเมสเสจไปให้ check_box ดังนี้

```
SendMessage (hCheckBox,
BM_  SETCHECK, 1, OL);
```

อย่างไรก็ตาม วินโดวส์ได้จัดเตรียมสไตล์ BS_AUTOCHECKBOX ซึ่งจะทำหน้าที่แสดงและไม่แสดงตัวเช็คโดยอัตโนมัติ

Radio Buttons

ใช้ในงานคล้าย ๆ กับ check box แต่สำหรับ radio button จะยอมให้ผู้ใช้เลือกออปชันได้ทางเดียวเท่านั้น (จากหลายทาง) ตัวอย่างการใช้งานได้แก่ ให้ผู้ใช้เลือกฟอร์แมตพารากราฟแบบชิดซ้าย, ชิดขวา หรือชิดซ้ายขวา ซึ่งผู้ใช้จะต้องเลือกอย่างใดอย่างหนึ่งเท่านั้น

สไตล์ของ radio Buttons คือ BS_RADIOBUTTON ส่วนการแสดงตัวเช็ค (วงกลมด้น) ทำเหมือนกับ check box หรือจะใช้ฟังก์ชัน Check-RadioButton

Owner-Draw Buttons

ทำหน้าที่คล้ายกับ button อื่น ๆ เราใช้ button นี้ เมื่อต้องการเปลี่ยนรูปร่างหน้าตาของ button รายละเอียดของ button ชนิดนี้อ่านได้จาก reference ของ SDK

Group Boxes

ไม่ใช่ button เป็นแต่เพียงสี่เหลี่ยมที่มีชื่อบอกเพื่อจัดกลุ่มบนคอนโทรลให้เหมาะสมและดูสวยงาม

Static

เป็นคอนโทรลที่ใช้แสดงข้อความหรือรูปภาพ มักใช้คู่กับคอนโทรลอื่น ๆ เพื่อบอกว่าคอนโทรลนั้นใช้ทำอะไร

สไตล์ของ list box	คำอธิบาย
LBS_BORDER	มีกรอบ
LBS_NOTIFY	ส่งเมสเสจให้วินโดว์พ่อด้วย
LBS_SORT	เรียงลำดับข้อความ
WS_VSCROLL	มี scroll bar

รูปที่ 4 สไตล์ของ list box

```
#define IDC_LISTBOX     300
HWND hListBox;

hListBox = CreateWindow ("Listbox", NULL,
                         LBS_STANDARD | WS_CHILD | WS_VISIBLE,
                         20, 40, 120, 56, hWnd, IDC_LISTBOX,
                         hInstance, NULL);
```

รูปที่ 5 ตัวอย่างการสร้าง list box

List Box

list box ใช้ในงานที่ต้องการแสดงรายการของสิ่งใดสิ่งหนึ่ง เช่น ชื่อไฟล์ เป็นต้น และผู้ใช้มีสิทธิ์เลือกได้มากกว่า 1 สิ่ง list box มีสไตล์มากมายหลายอย่าง ที่สำคัญแสดงในรูปที่ 4

ทั้ง 4 สไตล์ในรูปที่ 4 รวมอยู่ในสไตล์ LBS_STANDARD ตัวอย่างการสร้าง List box แสดงในรูปที่ 5

เราสามารถเพิ่มรายการใน list ด้วยการส่งเมสเสจ LB_ADDSTRING ไปให้ List Box และลดรายการด้วยการส่งเมสเสจ LB_DELETESTRING ไปให้ List Box นอกจากนี้ยังมีเมสเสจเกี่ยวกับ List Box อีกมากมายแต่จะไม่กล่าว ณ ที่นี้

List Box นับว่าเป็นคอนโทรลที่มีประโยชน์มากและมีหลายหลากสไตล์ให้ใช้เช่น แบบให้เลือกหลายรายการได้ แบบหลายคอลัมน์ หรือแบบ owner-draw เป็นต้น

Combo Boxes

เป็นคอนโทรลที่ประกอบด้วย List Box และ Edit หรือ Static พูดง่าย ๆ ก็คือ Combo Box ประกอบด้วยสองส่วน ได้แก่ ส่วนแรกเป็น List Box ซึ่งอาจจะ list box แบบปกติหรือแบบ dropdown ส่วนที่สองจะทำหน้าที่แสดงข้อความ ถ้าสามารถแก้ไขข้อความได้ ก็จะเป็น edit ถ้าแก้ไม่ได้ก็จะเป็น static

combo box มักใช้กับงานที่รายการประกอบด้วยหลาย ๆ สิ่ง และต้องการประหยัดเนื้อที่ โดยไม่แสดงรายการทั้งหมด

Edit

เป็นคอนโทรลที่ใช้ใส่และแก้ไขข้อความ edit มีหลากสไตล์ เช่น แบบใส่ข้อความได้มากกว่าหนึ่งบรรทัด แบบเลื่อนข้อความอัตโนมัติ แบบใช้เป็นที่รับพาสเวิร์ด เป็นต้น

Scroll Bars

เป็นคอนโทรลที่ใช้ในงานที่มีข้อมูลจำนวนมาก scroll bars จะใช้ในการเลื่อนดูข้อความเหล่านั้น สไตล์ของ scroll มี 2 อย่างคือ SBS_HORZ เป็น scroll bar ในแนวนอน และ SBS_VERT เป็น scroll bar ในแนวตั้ง

ที่กล่าวมานั้น ก็เป็นคอนโทรลชนิดต่าง ๆ และลักษณะการใช้งานปกติแล้วคอนโทรลเหล่านี้มักจะใช้ในไดอะล็อก-บ็อกซ์ ส่วนจะมีวิธีการสร้างไดอะล็อก-บ็อกซ์อย่างไรนั้น ท่านผู้อ่านจะได้ทราบในฉบับหน้า สำหรับฉบับนี้มีเพียงแค่นี้ครับ สวัสดีครับ